

Finite Length Weight Enumerator Analysis of Braided Convolutional Codes

Saeedeh Moloudi[†], Michael Lentmaier[‡], and Alexandre Graell i Amat[‡]

[†]Department of Electrical and Information Technology, Lund University, Lund, Sweden

[‡]Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden

{saeedeh.moloudi,michael.lentmaier}@eit.lth.se, alexandre.graell@chalmers.se

Abstract—Braided convolutional codes (BCCs) are a class of spatially coupled turbo-like codes (SC-TCs) with excellent belief propagation (BP) thresholds. In this paper we analyze the performance of BCCs in the finite block-length regime. We derive the average weight enumerator function (WEF) and compute the union bound on the performance for the uncoupled BCC ensemble. Our results suggest that the union bound is affected by poor distance properties of a small fraction of codes. By computing the union bound for the expurgated ensemble, we show that the floor improves substantially and very low error rates can be achieved for moderate permutation sizes. Based on the WEF, we also obtain a bound on the minimum distance which indicates that it grows linearly with the permutation size. Finally, we show that the estimated error floor for the uncoupled BCC ensemble is also valid for the coupled ensemble by proving that the minimum distance of the coupled ensemble is lower bounded by the minimum distance of the uncoupled ensemble.

I. INTRODUCTION

Low-density parity-check (LDPC) convolutional codes [1], also known as spatially coupled LDPC (SC-LDPC) codes [2], have attracted a lot of attention because they exhibit a threshold saturation phenomenon: the belief propagation (BP) decoder can achieve the threshold of the optimal maximum-a-posteriori (MAP) decoder. Spatial coupling is a general concept that is not limited to LDPC codes. Spatially coupled turbo-like codes (SC-TCs) are proposed in [3], where some block-wise spatially coupled ensembles of parallel concatenated codes (SC-PCCs) and serially concatenated codes (SC-SCCs) are introduced. Braided convolutional codes (BCCs) [4] are another class of SC-TCs. The original BCC ensemble has an inherent spatially coupled structure with coupling memory $m = 1$. Two extensions of BCCs to higher coupling memory, referred to as Type-I and Type-II BCCs, are proposed in [5].

The asymptotic behavior of BCCs, SC-PCCs and SC-SCCs is analyzed in [6] where the exact density evolution (DE) equations are derived for the binary erasure channel (BEC). Using DE, the thresholds of the BP decoder are computed for both uncoupled and coupled ensembles and compared with the corresponding MAP thresholds. The obtained numerical results demonstrate that threshold saturation occurs for all considered SC-TC ensembles if the coupling memory is large enough. Moreover, the occurrence of threshold saturation is proved analytically for SC-TCs over the BEC in [6], [7].

This work was supported in part by the Swedish Research Council (VR) under grant #621-2013-5477.

While the uncoupled BCC ensemble suffers from a poor BP threshold, the BP threshold of the coupled ensemble improves significantly even for coupling memory $m = 1$. Comparing the BP thresholds of SC-TCs in [6] indicates that for a given coupling memory, the Type-II BCC ensemble has the best BP threshold for almost all code rates.

Motivated by the good asymptotic performance and the excellent BP thresholds of BCCs, our aim in this paper is analyzing the performance of BCCs in the finite block-length regime by means of the ensemble weight enumerator. As a first step, we derive the finite block-length ensemble weight enumerator function (WEF) of the uncoupled ensemble by considering uniform random permutations. Then we compute the union bound for uncoupled BCCs. The unexpectedly high error floor predicted by the bound suggests that the bound is affected by the bad performance of codes with poor distance properties. We therefore compute the union bound on the performance of the expurgated ensemble by excluding the codes with poor distance properties. The expurgated bound demonstrates very low error floors for moderate permutation sizes. We also obtain a bound on the minimum distance of the BCC ensemble which reveals that the minimum distance grows linearly with the permutation size.

Finally, we prove that the codeword weights of the coupled ensemble are lower bounded by those of the uncoupled ensemble. Thus, the minimum distance of the coupled BCC ensemble is larger than the minimum distance of the uncoupled BCC ensemble. From this, we conclude that the estimated error floor of the uncoupled ensemble is also valid for the coupled ensemble.

II. COMPACT GRAPH REPRESENTATION OF TURBO-LIKE CODES

In this section, we describe three ensembles of turbo-like codes, namely PCCs, uncoupled BCCs, and coupled BCCs, using the compact graph representation introduced in [6].

A. Parallel Concatenated Codes

Fig. 1(a) shows the compact graph representation of a PCC ensemble with rate $R = \frac{N}{3N} = \frac{1}{3}$, where N is the permutation size. These codes are built of two rate-1/2 recursive systematic convolutional encoders, referred to as the upper and lower component encoder. The corresponding trellises are denoted by T^U and T^L , respectively. In the graph, factor nodes, represented

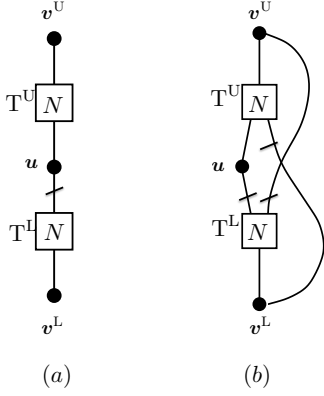


Fig. 1. (a) Compact graph representation of (a) PCCs (b) BCCs.

by squares, correspond to trellises. All information and parity sequences are shown by black circles, called variable nodes. The information sequence, \mathbf{u} , is connected to factor node T^U to produce the upper parity sequence \mathbf{v}^U . Similarly, a reordered copy of \mathbf{u} is connected to T^L to produce \mathbf{v}^L . In order to emphasize that a reordered copy of \mathbf{u} is used in T^L , the permutation is depicted by a line that crosses the edge which connects \mathbf{u} to T^L .

B. Braided Convolutional Codes

1) *Uncoupled BCCs*: The original BCCs are inherently a class of SC-TCs [3], [4], [5]. An uncoupled BCC ensemble can be obtained by tailbiting a BCC ensemble with coupling length $L = 1$. The compact graph representation of this ensemble is shown in Fig. 1(b). The BCCs of rate $R = \frac{1}{3}$ are built of two rate-2/3 recursive systematic convolutional encoders. The corresponding trellises are denoted by T^U and T^L , and referred to as the upper and lower trellises, respectively. The information sequence \mathbf{u} and a reordered version of the lower parity sequence \mathbf{v}^L are connected to T^U to produce the upper parity sequence \mathbf{v}^U . Likewise, a reordered version of \mathbf{u} and a reordered version of \mathbf{v}^U are connected to T^L to produce \mathbf{v}^L .

2) *Coupled BCCs, Type-I*: Fig. 2(a) shows the compact graph representation of the original BCC ensemble, which can be classified as Type-I BCC ensemble [5] with coupling memory $m = 1$. As depicted in Fig. 2(a), at time t , the information sequence \mathbf{u}_t and a reordered version of the lower parity sequence at time $t - 1$, \mathbf{v}_{t-1}^L , are connected to T_t^U to produce the current upper parity sequence \mathbf{v}_t^U . Likewise, a reordered version of \mathbf{u}_t and \mathbf{v}_{t-1}^U are connected to T_t^L to produce \mathbf{v}_t^L . At time t , the inputs of the encoders come only from time t and $t - 1$, hence the coupling memory is $m = 1$.

3) *Coupled BCCs, Type-II*: Fig. 2(b) shows the compact graph representation of Type-II BCCs with coupling memory $m = 1$. As depicted in the figure, in addition to the coupling of the parity sequences, the information sequence is also coupled. At time t , the information sequence \mathbf{u}_t is divided into two sequences $\mathbf{u}_{t,0}$ and $\mathbf{u}_{t,1}$. Likewise, a reordered copy of the information sequence, $\tilde{\mathbf{u}}_t$, is divided into two sequences $\tilde{\mathbf{u}}_{t,0}$ and $\tilde{\mathbf{u}}_{t,1}$. At time t , the first inputs of the upper and lower encoders are reordered versions of the sequences

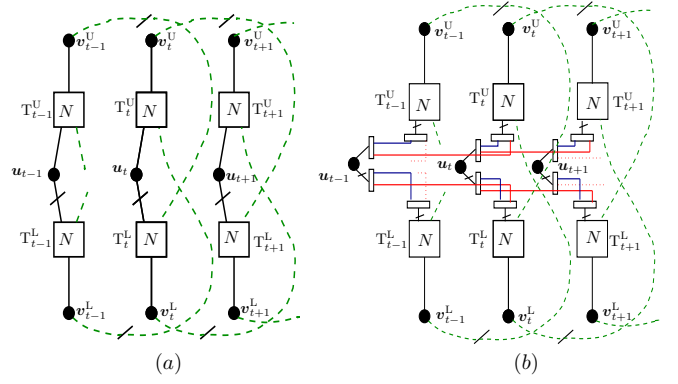


Fig. 2. Compact graph representation of coupled BCCs with coupling memory $m = 1$ (a) Type-I (b) Type-II.

$(\mathbf{u}_{t,0}, \mathbf{u}_{t-1,1})$ and $(\tilde{\mathbf{u}}_{t,0}, \tilde{\mathbf{u}}_{t-1,1})$, respectively.

III. INPUT-PARITY WEIGHT ENUMERATOR

A. Input-Parity Weight Enumerator for Convolutional Codes

Consider a rate-2/3 recursive systematic convolutional encoder. The input-parity weight enumerator function (IP-WEF), $A(I_1, I_2, P)$, can be written as

$$A(I_1, I_2, P) = \sum_{i_1} \sum_{i_2} \sum_p A_{i_1, i_2, p} I_1^{i_1} I_2^{i_2} P^p,$$

where $A_{i_1, i_2, p}$ is the number of codewords with weights i_1 , i_2 , and p for the first input, the second input, and the parity sequence, respectively.

To compute the IP-WEF, we can define a transition matrix between trellis sections denoted by \mathbf{M} . This matrix is a square matrix whose element in the r th row and the c th column $[\mathbf{M}]_{r,c}$ corresponds to the trellis branch which starts from the r th state and ends up at the c th state. More precisely, $[\mathbf{M}]_{r,c}$ is a monomial $I_1^{i_1} I_2^{i_2} P^p$, where i_1 , i_2 , and p can be zero or one depending on the branch weights.

For a rate-2/3 convolutional encoder with generator matrix

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 1/7 \\ 0 & 1 & 5/7 \end{pmatrix}, \quad (1)$$

in octal notation, the matrix \mathbf{M} is

$$\mathbf{M}(I_1, I_2, P) = \begin{pmatrix} 1 & I_2 P & I_1 I_2 & I_1 P \\ I_1 & I_1 I_2 P & I_2 & P \\ I_2 P & 1 & I_1 P & I_1 I_2 \\ I_1 I_2 P & I_1 & P & I_2 \end{pmatrix}. \quad (2)$$

Assume termination of the encoder after N trellis sections. The IP-WEF can be obtained by computing \mathbf{M}^N . The element $[\mathbf{M}^N]_{1,1}$ of the resulting matrix is the corresponding IP-WEF.

The WEF of the encoder is defined as

$$A(W) = \sum_{w=1}^N A_w W^w = A(I_1, I_2, P)|_{I_1=I_2=P=W},$$

where A_w is the number of codewords of weight w .

In a similar way, we can obtain the matrix M for a rate-1/2 convolutional encoder. The IP-WEF of the encoder is $[M^N]_{1,1}$ and is given by

$$A(I, P) = \sum_i \sum_p A_{i,p} I^i P^p,$$

where $A_{i,p}$ is the number of codewords of input weight i and parity weight p .

B. Parallel Concatenated Codes

For the PCC ensemble in Fig. 1(a), the IP-WEFs of the upper and lower encoders are defined by $A^{Tu}(I, P)$ and $A^{Tl}(I, P)$, respectively. The IP-WEF of the overall encoder, $A^{PCC}(I, P)$, depends on the permutation that is used, but we can compute the average IP-WEF for the ensemble. The coefficients of the IP-WEF of the PCC ensemble [8] can be written as

$$\bar{A}_{i,p}^{PCC} = \frac{\sum_{p_1} A_{i,p_1}^{Tu} \cdot A_{i,p-p_1}^{Tl}}{\binom{N}{i}}. \quad (3)$$

C. Braided Convolutional Codes

For the uncoupled BCC ensemble depicted in Fig. 1(b), the IP-WEFs of the upper and lower encoders are denoted by $A^{Tu}(I_1, I_2, P)$ and $A^{Tl}(I_1, I_2, P)$, respectively. To derive the average WEF, we have to average over all possible combinations of permutations. The coefficients of the IP-WEF of the uncoupled BCC ensemble can be written as

$$\bar{A}_{i,p}^{BCC} = \frac{\sum_{p_1} A_{i,p_1}^{Tu} \cdot A_{i,p-p_1}^{Tl}}{\binom{N}{i} \binom{N}{p_1} \binom{N}{p-p_1}}. \quad (4)$$

Remark: It is possible to interpret the BCCs in Fig. 1(b) as protograph-based generalized LDPC codes with trellis constraints. As a consequence, the IP-WEF of the ensemble can also be computed by the method presented in [9], [10].

IV. PERFORMANCE BOUNDS FOR BRAIDED CONVOLUTIONAL CODES

A. Bounds on the Error Probability

Consider the PCC and BCC ensembles in Fig. 1 with permutation size N . For transmission over an additive white Gaussian noise (AWGN) channel, the bit error rate (BER) of the code is upper bounded by

$$P_b \leq \sum_{i=1}^N \sum_{p=1}^{2N} \frac{i}{N} \bar{A}_{i,p} Q \left(\sqrt{2(i+p)R \frac{E_b}{N_0}} \right), \quad (5)$$

and the frame error rate (FER) is upper bounded by

$$P_F \leq \sum_{i=1}^N \sum_{w=1}^{2N} \bar{A}_{i,p} Q \left(\sqrt{2(i+p)R \frac{E_b}{N_0}} \right), \quad (6)$$

where $Q(\cdot)$ is the Q -function and $\frac{E_b}{N_0}$ is the signal-to-noise ratio.

The truncated union bounds on the BER and FER of the PCC ensemble in Fig. 1(a) are shown in Fig. 3. We have considered identical component encoders with generator matrix $G = (1, 5/7)$ in octal notation and permutation size

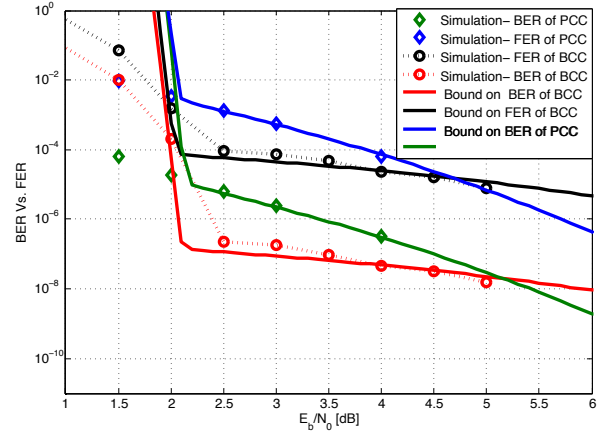


Fig. 3. Simulation results and bound on performance of the PCC and BCC ensembles.

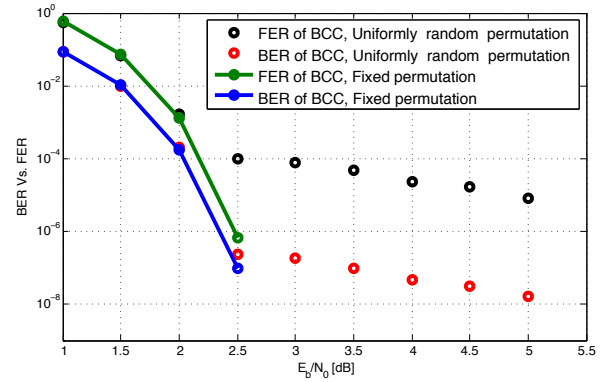


Fig. 4. Simulation results for BCC with uniformly random permutations and fixed permutations.

$N = 512$. We also plot the bounds for the uncoupled BCC ensemble with identical component encoders with generator matrix given in (1). The bounds are truncated at a value greater than the corresponding Gilbert-Varshamov bound.

Simulation results for the PCC and the uncoupled BCC ensemble are also provided in Fig. 3. To simulate the average performance, we have randomly selected new permutations for each simulated block. The simulation results are in agreement with the bounds for both ensembles. It is interesting to see that the error floor for the BCC ensemble is quite high and the slope of the floor is even worse than that of the PCC ensemble.

Fig. 4 shows simulation results for BCCs with randomly selected but fixed permutations. According to the figure, for the BCC with fixed permutations, the performance improves and no error floor is observed. For example, at $\frac{E_b}{N_0} = 2.5$ dB, the FER improves from $9.5 \cdot 10^{-5}$ to $6.8 \cdot 10^{-7}$. Comparing the simulation results for permutations selected uniformly at random and fixed permutations, suggests that the bad performance of the BCC ensemble is caused by a fraction of codes with poor distance properties. In the next subsection, we demonstrate that the performance of BCCs improves significantly if we use expurgation.

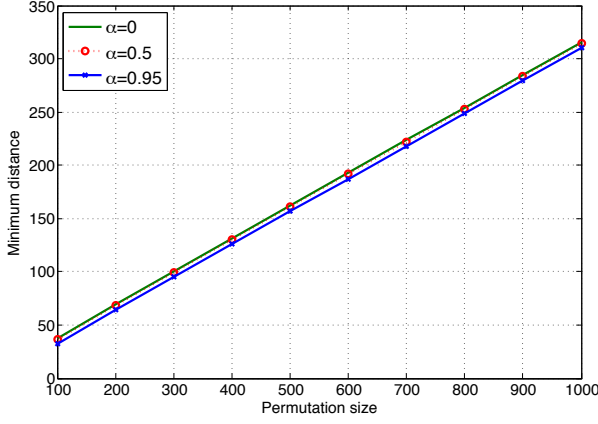


Fig. 5. Bound on the minimum distance for the BCC ensemble.

B. Bound on the Minimum Distance and Expurgated Union Bound

Using the average WEF, we can derive a bound on the minimum distance. We assume that all codes in the ensemble are selected with equal probability. Therefore, the total number of codewords of weight w over all codes in the ensemble is $N_c \cdot \bar{A}_w$, where N_c is the number of possible codes. As an example, N_c is equal to $(N!)^3$ for the BCC ensemble.

Assume that

$$\sum_{w=1}^{\hat{d}-1} \bar{A}_w < 1 - \alpha, \quad (7)$$

for some integer value $\hat{d} > 1$ and a given α , $0 \leq \alpha < 1$. Then a fraction α of the codes cannot contain codewords of weight $w < \hat{d}$. If we exclude the remaining fraction $1 - \alpha$ of codes with poor distance properties, the minimum distance of the remaining codes is lower bounded by $d_{\min} \geq \hat{d}$.

The best bound can be obtained by computing the largest \hat{d} that satisfies the condition in (7). Considering \bar{A}_w for different permutation sizes, this bound is shown in Fig. 5 for $\alpha = 0$, $\alpha = 0.5$, and $\alpha = 0.95$. According to the figure, the minimum distance of the BCC ensemble grows linearly with the permutation size. The bound corresponding to $\alpha = 0.95$, which is obtained by excluding only 5% of the codes, is very close to the existence bound for $\alpha = 0$. This means that only a small fraction of the permutations leads to poor distance properties.

Excluding the codes with $d_{\min} < \hat{d}$, the BER of the expurgated ensemble is upper bounded by

$$P_b \leq \frac{1}{\alpha} \sum_{i=1}^{kN} \sum_{\substack{p=1 \\ i+p \geq \hat{d}}}^{(n-k)N} \frac{i}{N} \bar{A}_{i,p} Q \left(\sqrt{2(i+p)R \frac{E_b}{N_0}} \right). \quad (8)$$

For the BCC ensemble, the expurgated bounds on the BER are shown in Fig. 6 for $\alpha = 0.5$ and permutation sizes $N = 128, 256$, and 512 . The error floors estimated by the expurgated bounds are much steeper than those given by the unexpurgated bounds. The expurgated bounds on the BER are also shown in

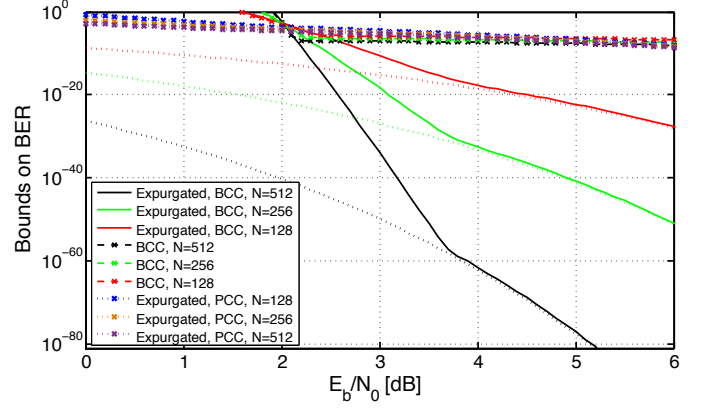


Fig. 6. Expurgated union bound on the performance of PCC and BCC.

Fig. 6 for the PCC ensemble. These bounds demonstrate that expurgation does not improve the performance of the PCC ensemble significantly.

V. SPATIALLY COUPLED BRAIDED CONVOLUTIONAL CODES

The performance of BCCs in the waterfall region can be significantly improved by spatial coupling. To demonstrate it, we provide simulation results for the uncoupled BCCs and Type-II BCCs for $N = 1000$ and 5000 . For coupled BCCs, we consider coupling length $L = 100$ and a sliding window decoder with window size $W = 5$ [11]. For all cases, the permutations are selected randomly but fixed. Simulation results are shown in Fig. 7. According to the figure, for a given permutation size, Type-II BCCs perform better than uncoupled BCCs. As an example, for $N = 5000$, the performance improves almost 1.5 dB. We also compare the uncoupled and coupled BCCs with equal decoding latency. In this case, we consider $N = 5000$ and $N = 1000$ for the uncoupled and coupled BCCs, respectively. Considering equal decoding latency, the performance of the coupled Type-II BCCs is still significantly better than that of the uncoupled BCCs.

The coupled BCCs have good performance in the waterfall region and their error floor is so low that it cannot be observed. It is possible to generalize equation (4) for the coupled BCC ensembles in Fig. 2 but the computational complexity is significantly increased. In the following theorem, we establish a connection between the WEF of the uncoupled BCC ensemble and that of the coupled ensemble. More specifically, we show that the weights of codewords cannot decrease by spatial coupling. A similar property is shown for LDPC codes in [12], [13], [14].

Theorem 1: Consider an uncoupled BCC $\tilde{\mathcal{C}}$ with permutations Π , Π^U and Π^L . This code can be obtained by means of tailbiting an original (coupled) BCC \mathcal{C} with time-invariant permutations $\Pi_t = \Pi$, $\Pi_t^U = \Pi^U$ and $\Pi_t^L = \Pi^L$. Let $\mathbf{v} = (\mathbf{v}_1, \dots, \mathbf{v}_t, \dots, \mathbf{v}_L)$, $\mathbf{v}_t = (\mathbf{u}_t, \mathbf{v}_t^U, \mathbf{v}_t^L)$, be an arbitrary code sequence of \mathcal{C} . Then there exists a codeword $\tilde{\mathbf{v}} \in \tilde{\mathcal{C}}$ that satisfies

$$w_H(\tilde{\mathbf{v}}) \leq w_H(\mathbf{v}),$$

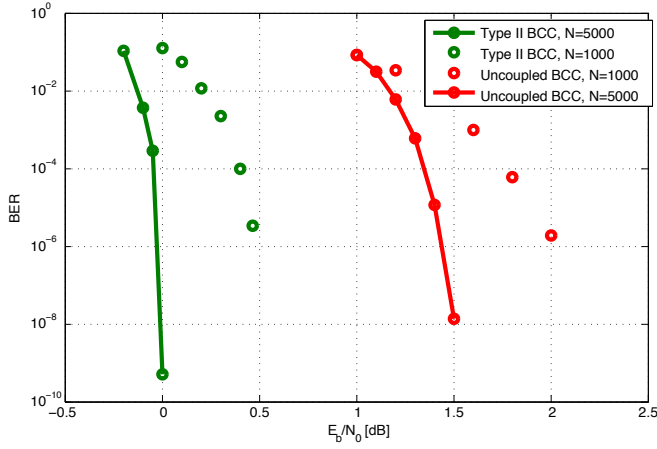


Fig. 7. Simulation results for uncoupled and coupled BCCs with fixed permutations, $N = 1000$ and $N = 5000$.

i.e., the coupling does either preserve or increase the Hamming weight of valid code sequences.

Proof: A valid code sequence of \mathcal{C} has to satisfy the local constraints

$$(\mathbf{u}_t \quad \mathbf{v}_{t-1}^L \cdot \Pi_t^U \quad \mathbf{v}_t^U) \cdot \mathbf{H}_U^T = \mathbf{0} \quad (9)$$

$$(\mathbf{u}_t \cdot \Pi_t \quad \mathbf{v}_{t-1}^U \cdot \Pi_t^L \quad \mathbf{v}_t^L) \cdot \mathbf{H}_L^T = \mathbf{0} \quad (10)$$

for all $t = 1, \dots, L$, where \mathbf{H}_U and \mathbf{H}_L are the parity-check matrices that represent the constraints imposed by the trellises of the upper and lower component encoders, respectively. Since these constraints are linear and time-invariant, it follows that any superposition of vectors $\mathbf{v}_t = (\mathbf{u}_t, \mathbf{v}_t^U, \mathbf{v}_t^L)$ from different time instants $t \in \{1, \dots, L\}$ will also satisfy (9) and (10). In particular, if we let

$$\tilde{\mathbf{u}} = \sum_{t=1}^L \mathbf{u}_t, \quad \tilde{\mathbf{v}}^L = \sum_{t=1}^L \mathbf{v}_t^L, \quad \tilde{\mathbf{v}}^U = \sum_{t=1}^L \mathbf{v}_t^U,$$

then

$$(\tilde{\mathbf{u}} \quad \tilde{\mathbf{v}}^L \cdot \Pi^U \quad \tilde{\mathbf{v}}^U) \cdot \mathbf{H}_U^T = \mathbf{0} \quad (11)$$

$$(\tilde{\mathbf{u}} \cdot \Pi \quad \tilde{\mathbf{v}}^U \cdot \Pi^L \quad \tilde{\mathbf{v}}^L) \cdot \mathbf{H}_L^T = \mathbf{0}. \quad (12)$$

Here we have implicitly made use of the fact that $\mathbf{v}_t = \mathbf{0}$ for $t < 1$ and $t > L$. But now it follows from (11) and (12) that $\tilde{\mathbf{v}} = (\tilde{\mathbf{u}}, \tilde{\mathbf{v}}^U, \tilde{\mathbf{v}}^L) \in \tilde{\mathcal{C}}$, i.e., we obtain a codeword of the uncoupled code. If all non-zero symbols within \mathbf{v}_t occur at different positions for $t = 1, \dots, L$, then $w_H(\tilde{\mathbf{v}}) = w_H(\mathbf{v})$. If, on the other hand, the support of non-zero symbols overlaps, the weight of $\tilde{\mathbf{v}}$ is reduced accordingly and $w_H(\tilde{\mathbf{v}}) < w_H(\mathbf{v})$. ■

Corollary 1: The minimum distance of the coupled BCC \mathcal{C} is larger than or equal to the minimum distance of the uncoupled BCC $\tilde{\mathcal{C}}$,

$$d_{\min}(\mathcal{C}) \geq d_{\min}(\tilde{\mathcal{C}}).$$

□

From *Corollary 1*, we can conclude that the estimated floor for the uncoupled BCC ensemble is also valid for the coupled BCC ensemble.

VI. CONCLUSION

The finite block length analysis of BCCs performed in this paper, together with the DE analysis in [6], show that BCCs are a very promising class of codes. They provide both close-to-capacity thresholds and very low error floors even for moderate block lengths. However, we would like to remark that the bounds on the error floor in this paper assume a maximum likelihood decoder. In practice, the error floor of the BP decoder may be determined by absorbing sets. This can be observed, for example, for some ensembles of SC-LDPC codes [15]. Therefore, it would be interesting to analyze the absorbing sets of BCCs.

REFERENCES

- [1] A. Jiménez Feltström and K.Sh. Zigangirov, "Periodic time-varying convolutional codes with low-density parity-check matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 2181–2190, Sep. 1999.
- [2] S. Kudekar, T.J. Richardson, and R.L. Urbanke, "Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 803–834, Feb. 2011.
- [3] S. Moloudi, M. Lentmaier, and A. Graell i Amat, "Spatially coupled turbo codes," in *Proc. 8th Int. Symp. on Turbo Codes and Iterative Inform. Process. (ISTC)*, Bremen, Germany, 2014.
- [4] W. Zhang, M. Lentmaier, K.Sh. Zigangirov, and D.J. Costello, Jr., "Braided convolutional codes: a new class of turbo-like codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 316–331, Jan. 2010.
- [5] M. Lentmaier, S. Moloudi, and A. Graell i Amat, "Braided convolutional codes - a class of spatially coupled turbo-like codes," in *Proc. Int. Conference on Signal Process. and Commun. (SPCOM)*, Bangalore, India, 2014.
- [6] S. Moloudi, M. Lentmaier, and A. Graell i Amat, "Spatially coupled turbo-like codes," submitted to *IEEE Trans. Inf. Theory*. [online]. Available <https://arxiv.org/pdf/1604.07315.pdf>, Dec. 2015.
- [7] S. Moloudi, M. Lentmaier, and A. Graell i Amat, "Threshold saturation for spatially coupled turbo-like codes over the binary erasure channel," in *Proc. Inf. Theory Workshop (ITW)*, Jeju, South Korea, Oct. 2015.
- [8] S. Benedetto and G. Montorsi, "Unveiling turbo codes: some results on parallel concatenated coding schemes," *IEEE Trans. Inf. Theory*, vol. 42, no. 2, pp. 409–428, Mar. 1996.
- [9] S. Abu-Surra, W.E. Ryan, and D. Divsalar, "Ensemble enumerators for protograph-based generalized LDPC codes," *IEEE Global Telecommunications Conference (GLOBECOM)*, Nov. 2007.
- [10] S. Abu-Surra, D. Divsalar, and W. E. Ryan, "Enumerators for protograph-based ensembles of LDPC and generalized LDPC codes," vol. 57, no. 2, pp. 858–886, Feb. 2011.
- [11] M. Zhu, D. G. M. Mitchell, M. Lentmaier, D. J. Costello, and B. Bai, "Window decoding of braided convolutional codes," in *Proc. Inf. Theory Workshop (ITW)*, Jeju, South Korea, Oct. 2015.
- [12] D. Truhachev, K. S. Zigangirov, and D. J. Costello, "Distance bounds for periodically time-varying and tail-biting LDPC convolutional codes," *IEEE Trans. on Inf. Theory*, vol. 56, no. 9, pp. 4301–4308, Sep. 2010.
- [13] D. G. M. Mitchell, A. E. Pusane, and D. J. Costello, "Minimum distance and trapping set analysis of protograph-based LDPC convolutional codes," *IEEE Trans. on Inf. Theory*, vol. 59, no. 1, pp. 254–281, Jan. 2013.
- [14] R. Smarandache, A. E. Pusane, P. O. Vontobel, and D. J. Costello, "Pseudocodeword performance analysis for ldpc convolutional codes," *IEEE Trans. on Inf. Theory*, vol. 55, no. 6, pp. 2577–2598, June 2009.
- [15] D. G. M. Mitchell, L. Dolecek, and D. J. Costello, "Absorbing set characterization of array-based spatially coupled LDPC codes," in *Proc. IEEE Int. Symp. on Inf. Theory (ISIT)*, Honolulu, HI, USA, June 2014.